RL-TR-95-285
Final Technical Report
January 1996

DISTRIBUTED INTERACTIVE SIMULATION (DIS) FOR TACTICAL C3I

PAR Government Systems Corporation (PGSC)

Kevin C. Trott

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

Rome Laboratory
Air Force Materiel Command
Rome, New York

This report has been reviewed by the Rome Laboratory Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be releasable to the general public, including foreign nations.

RL-TR-95-285 has been reviewed and is approved for publication.

APPROVED:

ALEX F. SISTI
Project Engineer

FOR THE COMMANDER:

JOSEPH CAMERA
Technical Director
Intelligence & Reconnaissance Directorate

If your address has changed or if you wish to be removed from the Rome Laboratory mailing list, or if the addressee is no longer employed by your organization, please notify Rome Laboratory/(IRAE), Rome NY 13441. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	January 1996	Final May 94 - Sep 95

4. TITLE AND SUBTITLE
DISTRIBUTED INTERACTIVE SIMULATION (DIS) FOR TACTICAL C3I

5. FUNDING NUMBERS
C - F30602-94-C-0107
PE - 62702F
PR - 4594
TA - 15
WU - L6

6. AUTHOR(S)
Kevin C. Trott

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
PAR Government Systems Corporation (PGSC)
PAR Technology Park, 8383 Seneca Turnpike
New Hartford NY 13413-1191

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Rome Laboratory/IRAE
32 Hangar Rd
Rome NY 13441-4114

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
RL-TR-95-285

11. SUPPLEMENTARY NOTES
Rome Laboratory Project Engineer: Alex F. Sisti/IRAE/(315) 330-4518

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)
PAR Government Systems Corporation (PGSC) assembled a local area Distributed Interactive Simulation (DIS) network at Rome Laboratory, using several commercial and government off-the-shelf software components, as well as software developed specifically for this effort. This network provides an initial step toward a common, distributed modeling and simulation infrastructure to support future Air Force C3I systems research and development, integration, and acquisition programs at Rome Laboratory, and it provides a foundation for future modeling and simulation technology development. The synthetic environment provided by this initial DIS network, once interfaced with the real and developmental C3I systems at Rome Laboratory, will allow these systems to be driven with realistic, dynamic simulated inputs, and will also allow the target nomination lists produced by these systems to carry out simulated strike missions within the synthetic environment.

14. SUBJECT TERMS
Distributed Interactive Simulation (DIS), C3I modeling and simulation, Air Force modeling and simulation, Time critical targets, Tactical C3I, Synthetic environment

15. NUMBER OF PAGES
60

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

This Final Technical Report documents the activities performed under the Distributed Interactive Simulation (DIS) for Tactical Command, Control, Communications, and Intelligence (C^3I) task, and summarizes the results of those activities. It is submitted by PAR Government Systems Corporation (PGSC) in accordance with CLIN 0002, CDRL Sequence No. A004 of Rome Laboratory contract F30602-94-C-0107.

Under this effort, PAR Government Systems Corporation (PGSC) assembled a local-area Distributed Interactive Simulation (DIS) network at Rome Laboratory, using several commercial and government off-the-shelf software components, as well as software developed specifically for this effort. This network provides an initial step toward a common, distributed modeling and simulation infrastructure to support future Air Force C^3I system research and development, integration, and acquisition programs at Rome Laboratory, as well as a foundation for future modeling and simulation technology development. This effort was specifically focused on providing DIS-based modeling and simulation support for improving the Air Force's ability to identify and prosecute various types of Time Critical Targets. The synthetic environment provided by this initial DIS network, once interfaced with the real and developmental C^3I systems at Rome Laboratory, will allow these systems to be driven with realistic, dynamic simulated inputs, and will also allow the target nomination lists produced by these systems to carry out simulated strike missions within the synthetic environment. This will allow the value of making near-real-time intelligence information available to Air Force C^3I systems to be demonstrated.

Section 2 provides background information on both the Time Critical Target problem and DIS technology. Section 3 describes the DIS for Tactical C^3I demonstration software system and its components, which consist of:

- the Observer Node – a collection of applications that provide access to DIS "ground truth" information, including:
  - a Plan View Display application, which displays ground truth on a map background,
  - a Stealth Vehicle Display application, which displays a perspective view of the simulated environment, and

  - a Data Logger application, which records and plays back the DIS Protocol Data Units (PDUs) that implement the exchange of ground truth information across the network,

- the Computer Generated Forces (CGF) Node – which simulates a variety of different types of ground and airborne platforms, individually or in small units,
- the Aircraft Node – which simulates friendly surveillance and strike aircraft, including AWACS, JSTARS, F-15s, and F-16s, and
- the Air Operations Center (AOC) Node – which simulates (in a highly abstract manner) the activities of an Air Force Air Operations Center.

Section 4 discusses the limitations of this demonstration software system, and identifies the lessons learned during its development. Section 5 contains recommendations for the future use of DIS technology by Rome Laboratory.

2. BACKGROUND

This section discusses the application problem that this effort attempted to address, and the key technologies used in attempting to address the problem. Section 2.1 discuses the Time Critical Target (TCT) problem, the WAR BREAKER strategy of attacking this problem by integrating operations and intelligence elements at all levels to improve situation awareness and battle management, and how RL's development of the Contingency Tactical Automated Planning System (CTAPS) architecture, and particularly the Rapid Application of Air Power (RAAP) system, fit into this strategy. Section 2.2 discusses modeling and simulation technology, and particularly Distributed Interactive Simulation (DIS) technology, and its potential role in the acquisition, development, and integration of the advanced C^3I systems needed to successfully attack the TCT problem. It also discusses how DIS technology can be integrated into Rome Laboratory's long history of using modeling and simulation to support the development of advanced tactical C^3I systems.

2.1 THE TIME CRITICAL TARGET PROBLEM

During Desert Storm, US forces achieved success through the exploitation of high technology weapons and the accelerated tempo of combat operations. However, review of those operations identified a major shortfall in the targeting and prosecution of time critical targets. While theater ballistic missiles received most of the publicity, time critical targets come in a wide variety of types, including mobile command and control centers, hidden sites (nuclear, biological or chemical), resupply and critical materials convoys, strike aircraft, and key ground units. Success against time critical targets requires highly developed situation awareness over a large geographical area, as well as timely assessment of enemy intentions, rapid detection, classification, and nomination of targets in "deep hide" with minimal false alarms, and quick, accurate targeting to support precision strikes. No single technology can provide an answer to this problem. Significant improvements are required in sensor coverage rates and effectiveness, intelligence quality and timeliness, and rapid and adaptive planning.

In spring of 1991, ARPA began a series of studies to look at the problem of time critical targets. The results of these initial studies led to the development of an integrated approach to the problem called WAR BREAKER. The objective of the WAR BREAKER

program was to develop and demonstrate advanced technologies and systems supporting synchronized, accurate prosecution of time critical fixed and mobile targets. The WAR BREAKER program consists of three major thrusts:

- Surveillance and Targeting (S&T) – with the objective of providing a capability for the rapid detection and classification of time critical targets through the development of a layered, integrated network of existing and advanced sensors and advanced sensor processing algorithms,
- Intelligence and Planning (I&P) – with the objective of providing a bridge between sensors and shooters to get inside the time critical target strike cycle by incrementally automating the targeting and planning process to provide distributed situation awareness and real-time battle management, and
- Systems Engineering and Evaluation – with the objective of integrating and controlling the resulting complex "system of systems", establishing technical and system level requirements for solving the time critical target problem through functional systems analysis, stochastic modeling, engineering simulations, and advanced distributed simulation, and enforcing the systems engineering discipline needed to maintain focus.

Figure 2-1 illustrates the WAR BREAKER concept of technology development in a mission driven context, which integrates the cycle of sensing & processing, correlation & analysis, planning, and battle management & attack execution. Sensing and processing provide basic data on the physical environment and enemy forces, while correlation and analysis convert this basic data into usable intelligence. Planning develops possible courses of action based on the current objectives, as well as on the situation of both enemy and friendly forces, and current weather and other environmental conditions. Once a specific course of action is selected, battle management controls the execution of the plan. The integration and automation of these processes will significantly reduce the time required to complete each cycle, improving the timeliness, accuracy, and completeness of both situation awareness and battle management, and allowing Time Critical Targets to be prosecuted much more effectively.

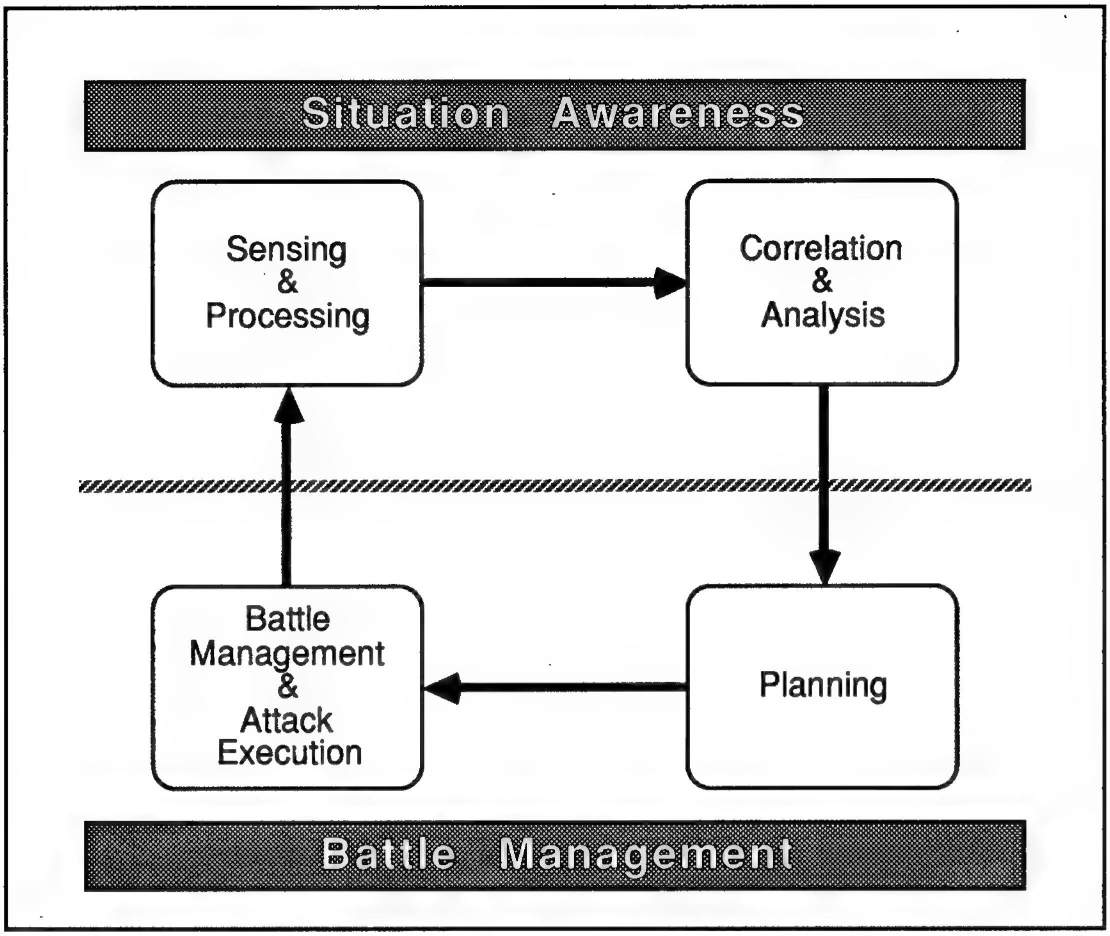

Figure 2-1. WAR BREAKER Mission Driven System Context

Rome Laboratory has long been aware of the importance of automating and integrating the intelligence and operations (planning, replanning, and execution) aspects of the air tasking cycle to address time critical targets, and has developed technology and systems which address several aspects of this problem. As shown in Figure 2-2, these include the Advanced Planning System (APS), addressing the mission planning needs of the Combat Plans Division; the Force Level Execution (FLEX) system, supporting the monitoring and control requirements of the Combat Operations Division; and the Rapid Application of Air Power (RAAP) system, supporting the situation analysis, target analysis, automated intelligence preparation of the battlefield, target nomination, and weaponeering functions of the Enemy Situation Correlation Division (ENSCD). Efforts to integrate these systems within the CTAPS architecture, based on a central intelligence database, are in progress.

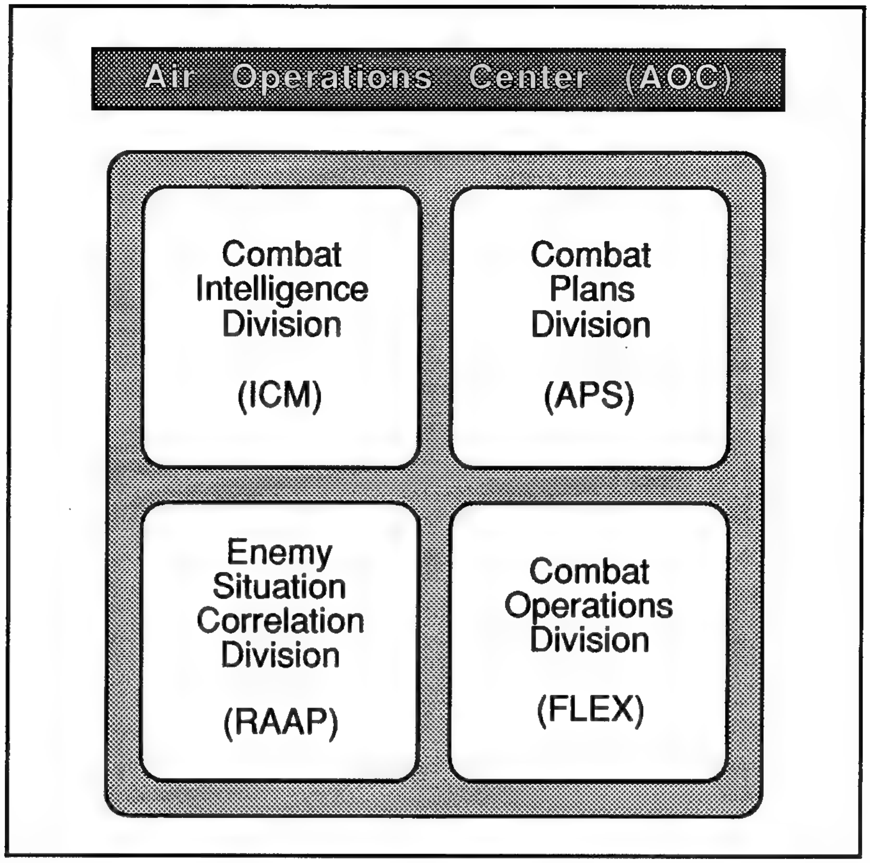

Figure 2-2. Air Operations Center (AOC) Organization

RAAP is an automated knowledge-based tool that has been designed to help integrate intelligence and operations processes. It is compliant with DoD Intelligence Information Systems (DoDIIS) standards through its use of the Military Intelligence Integrated Data System/Intelligence Data Base (MIIDS/IDB) database structure and data elements, as well as its use of TCP/IP networking standards, POSIX operating system standards, and X Window System and Motif user interface standards. As shown in Figure 2-3, RAAP performs the functions of situation analysis, automated intelligence preparation of the battlefield, target analysis, weaponeering, and target nomination. RAAP's primary source of information is the theater intelligence database. The primary outputs of RAAP include target nomination lists, which are sent to Combat Plans; immediate target attack messages, which are sent to Combat Operations; and immediate recce requests, which are sent to Collection Management. The benefits of RAAP include its role in facilitating the integration of operations and intelligence, reducing the ATO cycle, and better utilization of assets.

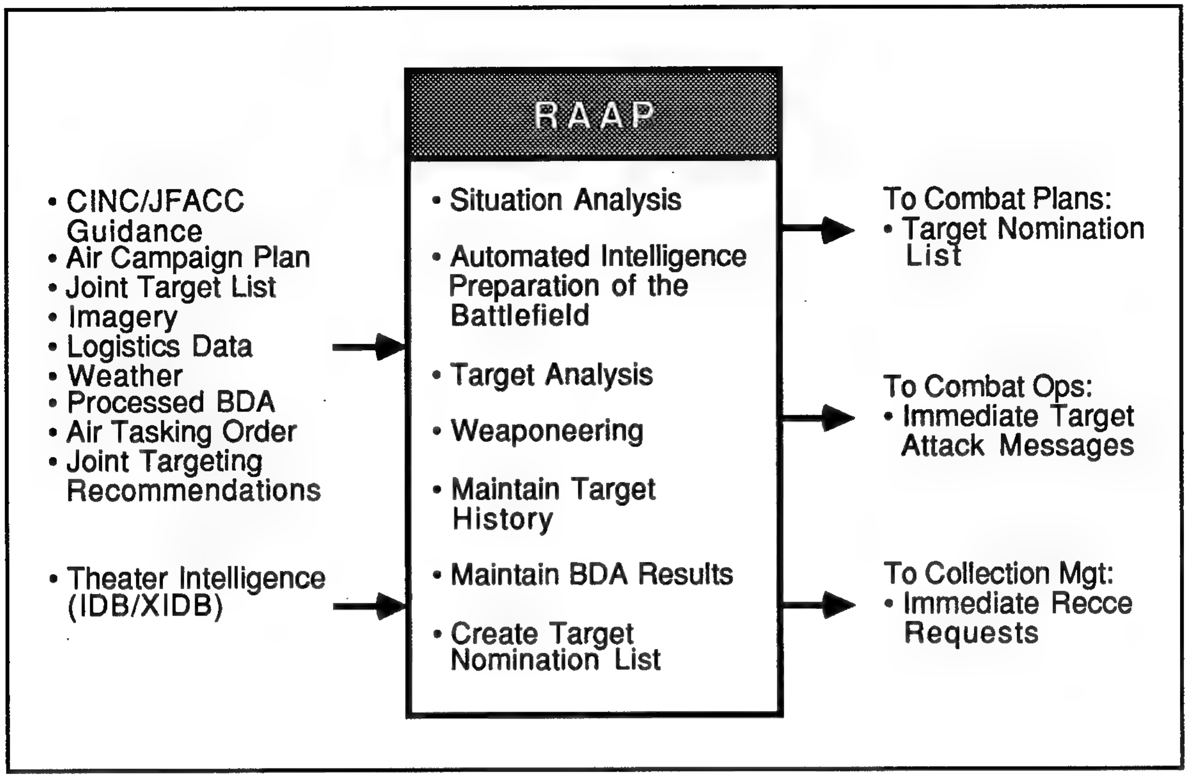

Figure 2-3. RAAP Functions, Inputs, and Outputs

Due to its charter and its long history of developing advanced systems for sensor exploitation, correlation/fusion, planning, and execution, Rome Laboratory is the logical conduit for the transitioning of technology developed under ARPA's WAR BREAKER program into existing and future Air Force C^3I systems. The evolution of the CTAPS architecture, including the integration of the APS, FLEX, and RAAP systems under the Operations/Intelligence Integration program, provides a vehicle and a context for this transitioning process. The effort described in this report was intended to support this arrangement through the exploitation of modeling and simulation technology, as described in the next section.

Because of its position at the boundary between the operations and intelligence processes, driving the RAAP system with simulated input, and using target nomination lists produced by RAAP to drive simulated air strikes, was chosen as the original focus for this effort. However, it was not possible to achieve this goal, due to conceptual difficulties resulting from the current CTAPS focus on fixed targets and not mobile, time critical targets, as well as practical issues such as terrain database compatibility.

2.2 DISTRIBUTED INTERACTIVE SIMULATION (DIS) TECHNOLOGY

Current Distributed Interactive Simulation (DIS) technology traces its roots back to the DARPA-sponsored Simulation Network (SIMNET) project, which began in 1983 and concluded in 1989. This R&D project successfully demonstrated the core technology required for networking large numbers of manned simulators, emulators, and computer generated forces (CGF). Dozens of networked M1 Abrams main battle tank and M2 Bradley infantry fighting vehicle simulators, as well as a small number of fixed and rotary wing aircraft simulators, and hundreds of CGF-controlled simulated vehicles, were linked together in a single synthetic battlefield environment. These simulators were located at eleven different sites in the US and Europe. The SIMNET project was extremely successful, particularly with respect to building a connection between the modeling and simulation community and the actual warfighters.

The success of the SIMNET program has contributed to increased recognition of the importance of modeling and simulation technology at all levels within DoD. This has resulted in the creation of the Defense Modeling and Simulation Office (DMSO), to coordinate DoD development and exploitation of modeling and simulation technology. It has also led to the growth of an officially sanctioned movement to develop a set of open standards for distributed simulation, based on the SIMNET networking protocols. This standards development effort, known as Distributed Interactive Simulation (DIS), is centered around a series of semiannual workshops coordinated and supported by the Institute for Simulation and Training (IST) of the University of Central Florida (UCF). Since the current work on DIS standards began in August 1989, the level of participation has grown steadily, with over 1000 people attending the most recent workshop in September 1995. In 1992, at the 14th Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) in San Antonio, more than 30 simulators, computer generated forces, and monitoring devices from more than 20 different organizations were linked together via Ethernet LAN in a demonstration of simulation interoperability supported by the DIS protocols. Simulated air, land, and naval forces operated together in a virtual world consisting of the area around Fort Hunter-Liggett in California and adjacent Pacific ocean waters. The 15th and 16th I/ITSECs, held in Orlando, were each attended by more than 7000 people. Participation in the DIS interoperability demonstrations has also increased

dramatically, with dozens of organizations simulating hundreds of entities including aircraft, helicopters, ships, and ground vehicles.

DIS sponsors within DoD include the Defense Modeling and Simulation Office (DMSO), the Advanced Research Projects Agency (ARPA), the US Army Simulation, Training and Instrumentation Command (STRICOM) (which has been designated the lead laboratory for development of DIS), the US Army Training and Doctrine Command (TRADOC), the Naval Air Systems Command (NAVAIR), the Naval Sea Systems Command (NAVSEA), the Air Force Air Combat Command (ACC), the Air Force Ballistic Missile Defense Organization (BMDO), the Air Force Training Special Program Office, and the US Special Operations Command (USSOCOM). Other supporting agencies include the Defense Information Systems Agency (DISA) (which is the DoD agent for developing information systems standards, and which will manage the Defense Simulation Internet (DSI)), and the National Security Agency, which is developing security procedures and encryption/decryption technology for use with DIS. A number of DoD programs, both large and small, are committed to using the DIS standards. These include the Advanced Distributed Simulation Technology (ADST) program, supported by both STRICOM and ARPA; STRICOM's Combined Arms Tactical Trainer (CATT) family of programs; the Navy's Battle Force Tactical Trainer (BFTT), and ARPA's WAR BREAKER program.

The primary purpose of the DIS standards is to define an infrastructure for linking simulations of various types at multiple locations to create realistic, complex, virtual "worlds" for the simulation of highly interactive activities. This infrastructure brings together systems built for separate purposes, products from various vendors, and platforms from various services and permits them to interoperate. A goal of DIS is to support an integrated mixture of "virtual" simulations, "live" entities, and "constructive" simulations. "Virtual" simulations are the continuous, real-time, human-in-the-loop simulation that have been the historical core of DIS (such as the original SIMNET tank simulators), and the more advanced simulators being developed under the Army's Close Combat Tactical Trainer (CCTT) program, as well as the Air Force's Theater Air Command and Control Simulation Facility (TACCSF). "Live" simulations involve crews in real vehicles moving on instrumented ranges; such as the Army's National Training Center or the Air Force's Red Flag ranges at Nellis AFB. "Constructive" simulations are more abstract automated wargames that are used for theater-level staff

training exercise; such as the Army's Corps Battle Simulation (CBS) or the Air Force's Air Warfare Simulation (AWSIM).

The DIS infrastructure provides interface standards, communications architectures, management structures, and other elements necessary to transform heterogeneous simulations into unified seamless synthetic environments. The initial focus of DIS development has been on training, but DIS is also intended to address mission rehearsal, reconstruction and analysis of actual battles, definition of requirements for new systems, development of tactical doctrine to support the use of new systems, and prototype evaluation. DIS technology is also beginning to be applied to non-military applications, including entertainment, education, air traffic control, disaster preparedness, and medical applications.

DIS models the virtual battlefield as a collection of "entities" that interact with one another by means of "events" that they cause. These events may be detected by other entities and may have effects on them, which may in turn cause other events that affect other entities. The heart of DIS is a set of protocols that convey information about entities and events across a local or wide area network, connecting various simulation nodes; each of which is responsible for maintaining the status of some of the entities in the virtual world. DIS technology is based on the following design principles:

- **Object/Event Architecture** - Information about fixed (non-changing) objects in the virtual environment is assumed to be known to all simulations and need not be transmitted. Dynamic objects keep each other informed of their movements and the events that they cause through the transmission of Protocol Data Units (PDUs) that describe any changes in entity state information.
- **Autonomy of Simulation Nodes** - From the perspective of each simulation node, all events are broadcast and are available to all interested objects. The node at which the event was caused does not need to determine which other nodes may be interested in that event. Each receiving node is responsible for determining the effects of an event on the entities that it is simulating. This autonomy principle allows nodes to join or leave an exercise in progress without disrupting the simulation.
- **Transmission of "Ground Truth" Information** - Each node transmits the absolute truth about the (externally observable) state of the object(s) it

represents. The receiving nodes are solely responsible for determining whether their objects can perceive an event and whether they are affected by it. Degradation of information is performed by the receiving node in accordance with an appropriate model of sensor characteristics before being passed on to human operators or automated systems.

- **Transmission of State Change Information Only** - Nodes transmit only changes in the behavior of the entities that they are simulating. This is intended to minimize the unnecessary transmission and processing of data. If an entity continues to do the same thing (e.g. straight and level flight at constant speed), the update rate drops to a predetermined minimum level.

- **"Dead Reckoning" Algorithms to Extrapolate State Information Between Updates** - Each simulation node maintains a simplified representation of the (externally visible) state of all nearby entities, and extrapolates their last reported states until the next state update information arrives. The node simulating each entity is responsible for transmitting new state information before the discrepancy between its "ground truth" information and the extrapolated approximations generated by the other nodes becomes too large. In order to support this, each node must maintain dead reckoning models of each of its own entities, and must continually compare its own "ground truth" state for each entity with the corresponding dead reckoning model, in order to determine when it must transmit a new update. State updates include not only location and orientation information, but also velocity and acceleration vectors that support the extrapolation.

- **Simulation Time Constraints** - The DIS standards were developed to support human-in-the-loop simulations, primarily involving manned simulators of ground and air platforms. DIS simulations currently operate in "real-time", using a performance standard of 100 milliseconds. Interactions between weapon systems, sensors, and tactical communications systems often occur at much faster rates. These types of interactions can be supported by DIS provided the communications latency requirements can be met by the communications network being used. If there are no human operators involved, it is possible to scale up DIS simulation time rates to allow faster than real time

operation, again provided that the communications network can meet the latency requirements that this imposes.

Initial DIS standards development has focused on the definition of the information that must flow between networked simulations to make them interoperable. These definitions include the messages, called Protocol Data Units (PDUs) that are exchanged by simulation nodes, as well as rules governing the transmission and processing of PDUs. The initial version of the DIS standard was submitted to the IEEE and was approved on 17 March 1993 as IEEE Standard 1278. The current version of the standard defines several types of PDUs:

- Entity State PDU – describing the externally observable state of a particular entity, including location, orientation, velocity, acceleration, positions of any articulated and/or attached parts, and appearance.
- Fire PDU – describing the firing of a weapon, including the firing location & entity, munition type, etc.
- Detonation PDU – describing the detonation of a weapon, including the impact location, munition type, etc.
- Collision PDU – describing a collision between two entities, including the identifiers of the colliding entities, location, velocity, mass, etc.
- Transmitter PDU – describing the external characteristics of a signal, including the power, frequency, etc.
- Signal PDU – describing the internal content of a signal, either data or voice.
- Logistics PDU Family – describing events associated with resupply & repair.
- Simulation Management (SIMAN) PDU Family – a collection of PDUs that allow a simulation manager to create, start, pause, stop, terminate, and query simulated entities on other nodes of a DIS network.

Standards are also in development for the communications architecture needed to support DIS, security, management, synthetic environment representation (including dynamic changes to the environment), field instrumentation (of training and test ranges), performance measurement, and a taxonomy of fidelity descriptors.

Although DIS technology has been maturing rapidly over the past several years, there still remains much to be accomplished, and there are still opportunities for Rome Laboratory to make significant contributions to this technology.

The challenges that must be overcome in order for DIS to reach its full potential include:

1. Force aggregation/deaggregation – DIS currently addresses platform-level simulation, but for many purposes it is more appropriate to be able to represent forces at the unit level, as platoons, companies, flights, etc. The representation of the tactical state (posture, readiness, etc.) of units is essential to the interoperability of constructive simulations with DIS. Mechanisms to dynamically aggregate and deaggregate forces in response to changing fidelity requirements within a simulation are also challenges that remain to be addressed.
2. Very large numbers of entities – In order for DIS technology to be used to support theater-level exercises and experiments, it will be necessary to scale up DIS simulations to at least 100,000 entities, to create what ARPA refers to as a Synthetic Theater of War (STOW).
3. Dynamic terrain – DIS exercises currently are limited to a static environment database; there is no mechanism to allow for changes in the environment due to the actions of the participants (creating and/or destroying bridges, roads, buildings, etc.) or of natural events (rain, snow, etc.).
4. Atmospheric effects – The DIS standards do not currently include the effects of weather, smoke, and other atmospheric effects on military operations within the synthetic environment.
5. Mechanisms to plan, initialize, control, and debrief exercises – Requirements for scenario preparation, execution control, and post-execution review and analysis have just begun to be addressed.
6. Interoperability of Computer Generated Forces (CGFs) – CGF systems are needed to provide DIS exercises with opposing forces, supporting forces, and other forces, and to allow a small number of people to control large forces.

There is a great deal of work still necessary before CGFs will be able to adequately fill many of the roles required in large-scale, realistic simulations.

2.3 DOD SUPPORT OF MODELING AND SIMULATION

The integration of modeling and simulation support has become a priority throughout DoD. The Defense Science Board (DSB) study on modeling and simulation, conducted in the summer of 1992, recommended that:

"All labs, test facilities, training ranges, service schools and industry should be fully networked and made DIS compatible"

"DIS standards and protocols should be incorporated into all appropriate developments and procurements"

It also recommended specific areas for investment in advanced distributed simulation technologies and tools, including:

- simulation scalability,
- fully and semi-automated forces (friendly and enemy),
- reusable terrain and environmental databases,
- modeling and simulation construction support tools, and
- verification, validation and accreditation.

As a result of these recommendations, modeling and simulation has achieved increased levels of visibility throughout DoD. The Defense Modeling and Simulation Office (DMSO) was created by the Director of Defense Research and Engineering (DDR&E) to coordinate modeling and simulation efforts and encourage standardization and interoperability. The DoD Modeling and Simulation Master Plan was developed, and was approved on 17 October 1995. It identifies six primary objectives:

1. Develop a common technical framework for M&S.
2. Provide timely and authoritative representations of the natural environment.
3. Provide authoritative representations of systems.
4. Provide authoritative representations of human behavior.

5. Establish modeling and simulation infrastructure to meet developer and end-user needs.
6. Share the benefits of modeling and simulation.

The common technical framework is currently being addressed in three ways.

- the High Level Architecture (HLA), which characterizes individual simulations and "federations" of simulations in terms of the types of objects which they model and the interactions among those objects, and defines services for object management, time management, simulation management, etc. The initial draft of the HLA has been developed by the DMSO-created Architecture Management Group (AMG), which operates analogously to the Object Management Group (OMG).
- Conceptual Models of the Mission Space (CMMS), which is attempting to develop a "data dictionary" for each DoD mission area.
- Data Standardization, which is attempting to provide standardized attribute values for the "objects" defined in the HLA and CMMS.

Representations of the natural environment are being addressed through the establishment of Executive Agents for each environmental domain. The Defense Mapping Agency is the Executive Agent for terrain. The Navy is the Executive Agent for oceans, and the Air Force is the Executive Agent for atmosphere and space. Each Executive Agent is responsible for providing leadership and coordination of efforts to develop standards in the environmental domain for which they are responsible.

Representations of systems and of human behavior have not yet been given much attention. Modeling and simulation infrastructure efforts include efforts to develop repositories of models and data, to develop verification, validation, and accreditation (VV&A) processes, and to develop communications networks like the Defense Simulation Internet. Finally, efforts are underway to share the benefits of distributed simulation technology with education, entertainment, and other application areas.

ARPA has been sponsoring a number of large DIS exercises through the Synthetic Theater of War (STOW) program. The STOW-Europe demonstration, held in conjunction with the Atlantic Resolve exercise, showed that it was feasible to link geographically distributed aircraft, naval, and ground vehicle simulators within the

context of a large-scale joint exercise. Other large-scale DIS exercises have included Kernel Blitz and Prairie Warrior.

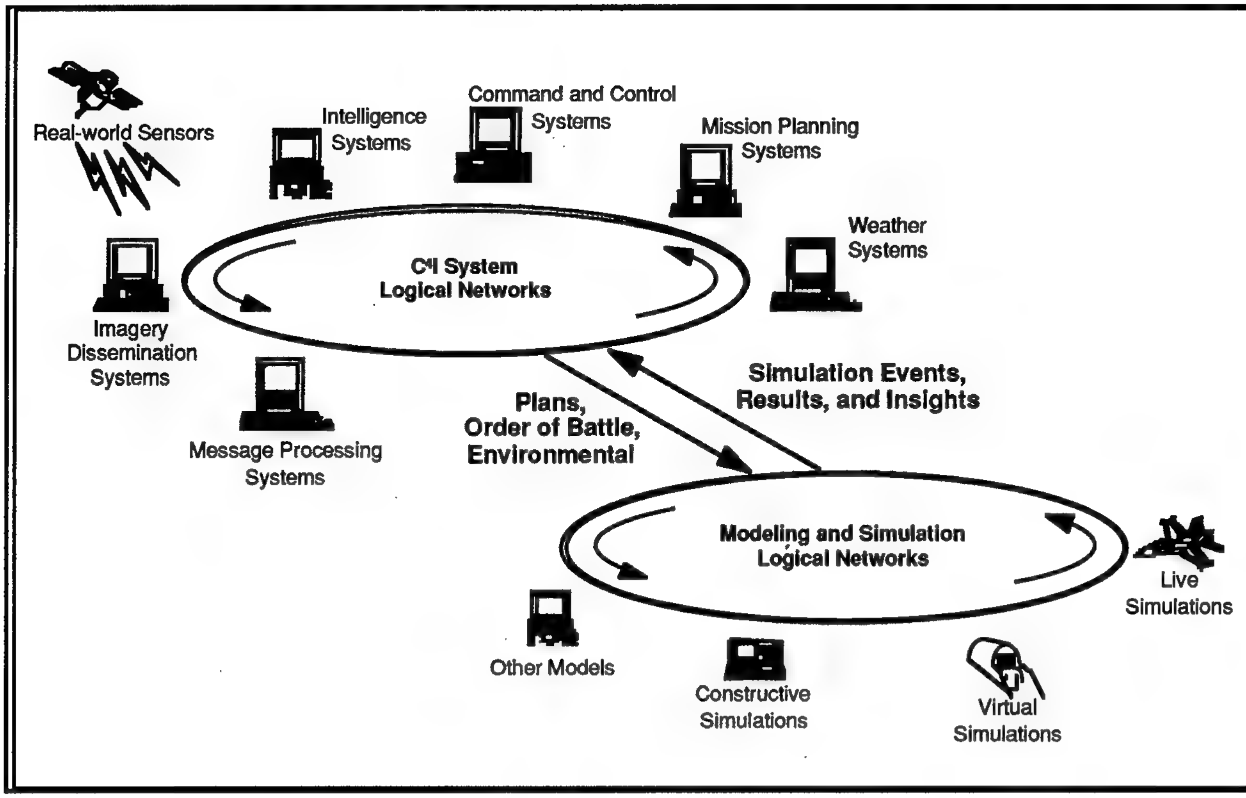

Figure 2-4. Linking Modeling & Simulation with Real C4I Systems

Another specific area in which there has been an increasing amount of interest and activity lately is the linking of modeling and simulation technology with real C4I systems. As shown in Figure 2-4, this is a bidirectional relationship. Simulations can be used to "drive" a C4I system, creating a simulated environment within which the C4I system can operate. Conversely, a C4I system can use a simulation to make predictions, extrapolating from the currently known state. In July 1995, a workshop on interfacing simulations and C4I systems was held at IDA. The programs represented at that workshop, which are listed in Table 2-1, included several programs involving the interfacing of the Air Force CTAPS system to various high-level simulation systems, such as the Air Warfare Simulation (AWSIM) and the Extended Air Defense Simulation (EADSIM). The COMPASS program, in which Rome Laboratory has also been involved to some degree, is adapting DIS technology to support distributed mission planning.

Table 2-1. M&S/C^4I Interoperability Efforts

Program	Gov't Organization	Systems Being Interfaced
SIMLINK	DISA/J8	JTLS – GCCS
JADS	DDSE&E	JSTARS/GSM – JANUS
KERNEL BLITZ	Navy	Link11/OTCIXF – ModSAF
RESA	NRaD	JOTS/NTDS – RESA
CFOR	ARPA	B2C2 – ModSAF
SRM	CECOM	SINGCARS model to system
JPSD	JPSD	ADOCS/ASAS – CLCGF
CWIC	Blue Flag	CTAPS – AWSIM
MASS	AF/ESC	CTAPS – EADSIM
Real Warrior	USAFE	CTAPS – AWSIM
COMPASS	NRaD	Mission Planners – DIS
ATTCS-CBS	AES/TEXCOM	ATTCS – CBS
ADSTE	FORCOM	MCE/TAOM/etc – STAGE

2.4 Air Force Support of Modeling and Simulation

In response to the Defense Science Board summer study recommendations in 1992, Air Force Material Command formed the Four Labs Modeling and Simulation for Science and Technology (FOURMOSST) working group, in which Rome Laboratory has participated, and also formed a Technical Planning Integrated Product Team (TPIPT). The Electronic Systems Center (ESC) has created the Modeling, Simulation, and Analysis Center (MASC). The Air Combat Command has created the Tactical Air Command and Control Simulation Facility (TACCSF) at Kirtland AFB. The National Air Intelligence Center (NAIC) has worked to improve its modeling and simulation capabilities under both the MASTER and PREFECT programs.

In 1993, an Air Force 4-Star Summit on Modeling and Simulation was held. As a result, the Air Force Modeling, Simulation, and Analysis Directorate (AF/XOM) was

created to coordinate Air Force modeling and simulation activities. Its initial focus was on the elimination of duplication of effort in the development of models and simulations. AF/XOM has been active in the development of the DIS standards, participating in the DIS User/Sponsor Committee and holding Air Force interest group meetings in conjunction with each DIS standards workshop.

In January 1995, GEN Ronald R. Fogleman, the Air Force Chief of Staff, ordered that a second Air Force 4-Star Summit on Modeling and Simulation be held in June 1995. This meeting resulted in "A New Vector" for Air Force modeling and simulation. The summit report contains the following statement by GEN Fogleman and Sheila E. Widnall, Secretary of the Air Force:

> "It is time to set a new vector for Air Force modeling and simulation. We need to expand our involvement and investment in advanced simulation technologies to improve our readiness and lower our costs today, and prepare us to dominate the battles of tomorrow."

Modeling and simulation within the Air Force supports two basic areas: analysis and training. The use of modeling and simulation to support analysis encompasses a wide variety of decision-making activities, ranging from basic research through test and evaluation to mission planning and rehearsal. Operational crews use modeling and simulation to make critical warfighting decisions. Acquisition programs use modeling and simulation to develop requirements and support funding decisions. Senior Air Force leadership use modeling and simulation to support force structuring decisions. Similarly, modeling and simulation is used throughout the Air Force for training of pilots, crews, and battlestaff. The improved decisions that result from the use of modeling and simulation for analysis, and the improved skills that result from the use of modeling and simulation for training, are both critical to the overall improvement of the Air Force's warfighting capability.

The Air Force 4-Star Summit on Modeling and Simulation produced a vision statement, illustrated in Figure 2-5, to be used to integrate the Air Force's modeling and simulation efforts. The key concept is that of a Joint Synthetic Battlespace, which can be considered to consist of an integrated modeling and simulation environment capable of combining many different types of simulations, ranging from detailed engineering models of specific equipment subsystems to highly aggregated

wargames, as well as manned simulators and pilots in live aircraft. It is also important to note that all of these models must be backed up by a library of standard data.

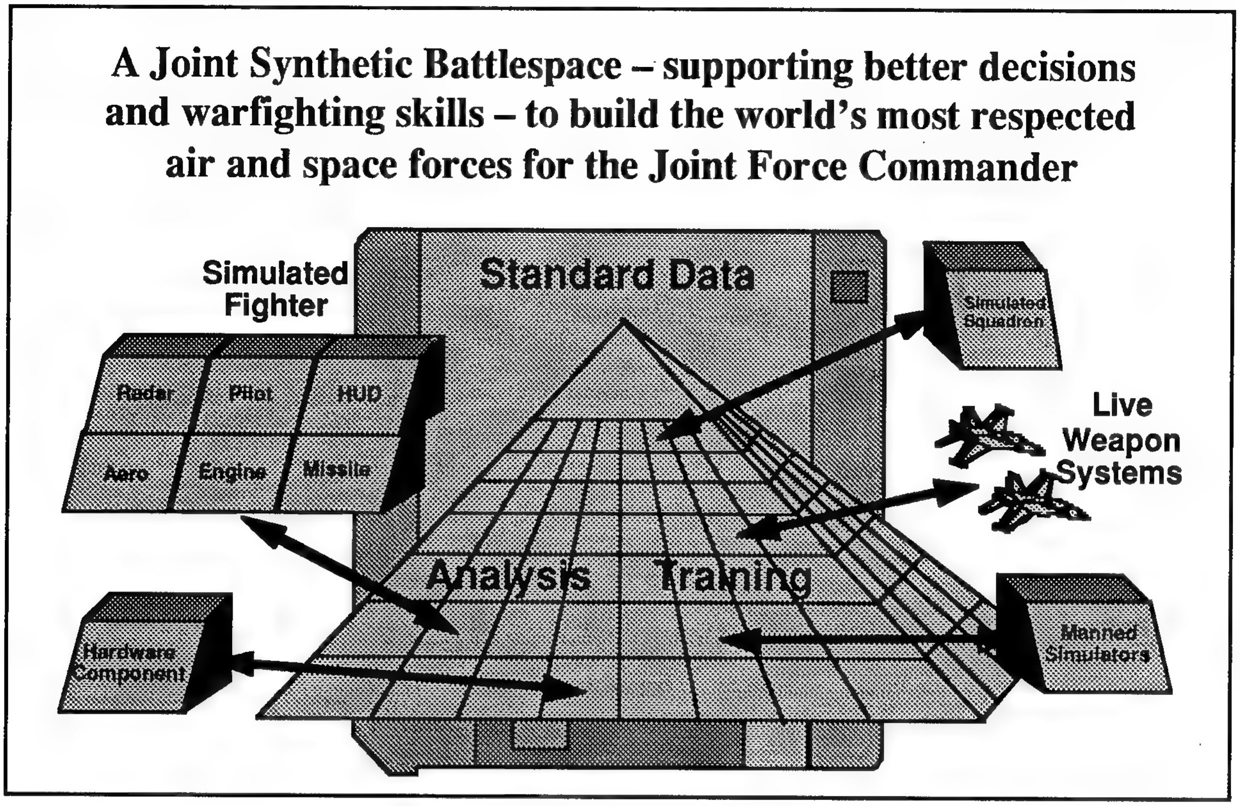

Figure 2-5. Air Force Vision for Modeling and Simulation Support

From a user's perspective, analysts, decision makers and warfighters must all be able to access the common battlespace from wherever they are currently located, whether that is in a laboratory, at a desk, in a cockpit simulator, in an Air Operations Center, or in an actual aircraft. When the Joint Synthetic Battlespace becomes a reality, it will allow the Air Force to achieve better decisions through analysis, and better skills through training, in an affordable manner based on a common modeling and simulation infrastructure. The goal of this is to provide the Joint Force Commander with forces capable of winning decisive victories with minimum loss of life.

The Directorate of Modeling, Simulation, and Analysis, Deputy Chief of Staff, Plans and Operations, Headquarters United States Air Force (AF/XOM) is the primary point of contact for modeling and simulation issues and activities within the Air Force, and represents the Air Force in joint, multi-service, and multi-agency modeling and simulation efforts. AF/XOM also provides leadership for the development of Air Force

modeling and simulation policy and resource strategy. AF/XOM is assisted by several other Air Force organizations and working groups, including:

- Air Force Studies and Analysis Agency (AFSAA), which is a Field Operating Agency reporting to AF/XOM, conducts analyses for Headquarters USAF and currently assists in implementing and supporting modeling and simulation policy.
- Air Force Simulation and Analysis Working Group (SAWG), which is an O-6 level working group with representatives from the key Air Staff and MAJCOM organizations involved with modeling and simulation, is the key forum for identifying modeling and simulation needs and opportunities and coordinating modeling and simulation policy.
- The Modeling and Simulation Technology Planning Integrated Product Team (M&S TPIPT) coordinates modeling and simulation activities with the Air Force Material Command, and provides technical support to operational commands and development programs.

In addition to these, the 4-Star Summit resulted in initial actions toward the creation of an Air Force Modeling and Simulation organization with the mission of supporting Air Force modeling and simulation users in the field and implementing Air Force modeling and simulation policies and initiatives. This organization will be located in Orlando, FL, with the Army's Simulation, Training, and Instrumentation Command (STRICOM), and the Naval Air Warfare Center – Training Systems Division (NAWC-TSD), and will be based on the Armstrong Laboratory operating detachment currently located there. This new organization will manage the implementation of the Air Force modeling and simulation roadmap, and will take on some of the modeling and simulation management responsibilities currently handled by AFSAA and other functional centers of expertise within the Air Force.

Also as a result of the 4-Star Summit, the Air Force is starting a number of other modeling and simulation initiatives, dealing with the areas of quality, people, and infrastructure. These include:

- The Air Force Modeling & Simulation Resource Repository (AFMSRR), which will consist of an on-line, distributed facility from which Air Force users can download models, data, and documentation. This will be accomplished in

cooperation with other DoD activities and will form a support node or subnetwork of an overall DoD network.

- The Air Force Verification, Validation, and Accreditation (VV&A) Program will provide partial funding for VV&A activities for key simulations and encouraging focus on fewer, more credible models and simulations. AFI 16-1001, currently in coordination, will define a complete V&V process followed by formal accreditation.

- The Prime Warrior training program prepares Air Force personnel for participation in joint wargames, exercises, and analyses. Its goal is to ensure that the Air Force participants understand the modeling and simulation and its limitations, and can ensure that air and space power is properly represented in these activities. Currently, this program is administered on an ad hoc basis by multiple Air Staff organizations, but a more formal program is being designed.

- Several modeling and simulation personnel initiatives, including:
  - Increasing the number of Air Force personnel with modeling and simulation experience through the use of educational courses,
  - Improved tracking of Air Force personnel with special or unique modeling and simulation experience,
  - A comprehensive review of all modeling and simulation related skills to ensure that personnel plans, policies, and programs exist to sustain future modeling and simulation requirements

The 4-Star Summit identified a need to provide effective manpower support to key modeling and simulation related organizations in the analysis and training communities, including additional manpower requirements within Field Operating Agencies, Centers, Major Command analytic staffs, and primary Joint M&S program offices. As validated requirements are identified, manpower will be reprogrammed from existing Air Force modeling and simulation activities.

- The Advanced M&S Connectivity Program will provide high-speed connectivity between Air Force installations to leverage existing DoD and Air Force programs, in conjunction with the Superhighway 2000 initiative. This program will provide additional hardware and software, including communications

security equipment, to allow Air Force facilities to connect to a high-speed, classified M&S distributed environment.

- The Air Combat Simulation Training Program will address current pilot training deficiencies and allow the Air Force to expand pilot training opportunities in spite of budget constraints through research and development to remove current simulator constraints, establishment of testbeds, and procurement and support of multiple networked simulators for each Air Force wing.
- Synthetic Battlespace for JFACC Training Program will provide a Joint Force Air Component Commander (JFACC) with a realistic synthetic battlespace, which will allow operators to train and exercise using their real-world C^4I equipment in a realistic wartime environment that reflects the entire range of Air Force operational capabilities.

In addition, the Air Force will participate in key Joint modeling and simulation programs, including:

- Joint Modeling and Simulation Integration Program (JMSIP) – an Air Force initiated program which will discourage ad hoc M&S development, create de facto standard models, and encourage development teaming, by allocating funds to maximize common efforts and target improvements based on an Air Force corporate assessment of their priority, as determined by a board of representatives from each Air Force command.
- DoD Simulation High Level Architecture (HLA) – a high-level simulation architecture that will be used to tie together models and simulations at various levels of detail.
- Modeling, Analysis, Simulation, and Training (MASTR) Database – an integrated, common source of data for analytic models, including a central database, and import and export software.
- Joint Modeling and Simulation System (J-MASS) – an Air Force directed program to develop a distributed, object-oriented M&S architecture and system for the more detailed, tactical level simulation models.
- National Air and Space Warfare Model (NASM) – an Air Force program focused on the development of a flexible framework for representing the full

range of air and space capabilities at the operational level to support battlestaff training. NASM will provide both a stand-alone capability, and the Air Force component of the Joint Simulation System.

- Joint Simulation System (JSIMS) – a distributed, object-oriented M&S architecture and system focused on the operational (campaign and mission) level for Joint battlestaff training.
- Joint Warfare Simulation (JWARS) – the analog of JSIMS for Joint campaign analysis, developing the next generation M&S architectures and systems for Joint analysis, as part of the Joint Analysis Model Improvement Program.

As the Air Force's laboratory for C^3I research and development, Rome Laboratory has an important role to play with respect to many of the initiatives and programs listed above. The development of C^3I concepts and systems at Rome Laboratory has always required modeling and simulation support. This has normally been provided within the scope of each individual program, resulting in considerable duplication of effort in the collection of data, the development of simulation models, and the development of realistic test and demonstration scenarios. Also, because there is little or no commonality across programs, it is seldom possible to directly compare the performance of different systems which perform similar functions. There is also no foundation for the integration of systems which perform complementary functions. In 1988, the Joint C3/IR Working Group for Enemy Force Simulation recognized that a common modeling and simulation support environment was needed to support the automation and integration of sensor data processing, correlation and fusion, intelligence processing, planning and execution. Such a simulation support environment is still needed by Rome Laboratory, and will be required in the near future, as new technologies and system concepts will be tested and evaluated within the context of a Joint Synthetic Battlespace before key development decisions are made.

3. SYSTEM DESCRIPTION

This section provides a description of the demonstration software system delivered under the DIS for Tactical C^3I contract, as installed in the Rome Laboratory ICARUS facility. Section 3.1 describes the overall organization of the demonstration software system. Section 3.2 describes the Observer Node, which includes the Stealth Vehicle Display application, the Plan View Display application, and the Data Logger application. The Computer Generated Forces (CGF) Node software is described in Section 3.3. Section 3.4 describes the Aircraft Node software, and Section 3.5 describes the Air Operations Center (AOC) Node software. Section 3.6 discusses the databases and files that are also essential components of the system.

3.1 SYSTEM CONFIGURATION

The DIS for Tactical C^3I demonstration software system consists of the following components:

- the Observer Node, which does not simulate any entities, but which allows "ground truth" information to be displayed, recorded, and played back,
- the Computer Generated Forces (CGF) Node, which simulates a wide variety of friendly and enemy ground forces, helicopters, and aircraft,
- the Aircraft Node, which simulates friendly surveillance (E-3 and E-8) and strike (F-15 and F-16) aircraft, and
- Air Operations Center (AOC) Node, which simulates an Air Operations Center in a very simplified, abstract manner.

It is possible to run multiple copies of the Observer Node applications, the CGF Node application, the Aircraft node application, and the AOC Node application simultaneously as part of a single DIS network exercise.

The organization of these components within a DIS local area network is shown in Figure 3-1. The Observer Node Stealth Vehicle Display and Data Logger applications run only on SGI workstations running the IRIX 5.2 or 5.3 operating system. The Observer Node Plan View Display application, and the CGF Node, Aircraft, and AOC Node applications can be run either on an SGI workstation under IRIX 5.2 or 5.3, or on a Sun workstation under the SunOS 4.1.3 operating system. Each application transmits and/or receives standard DIS Protocol Data Units (PDUs) over the network.

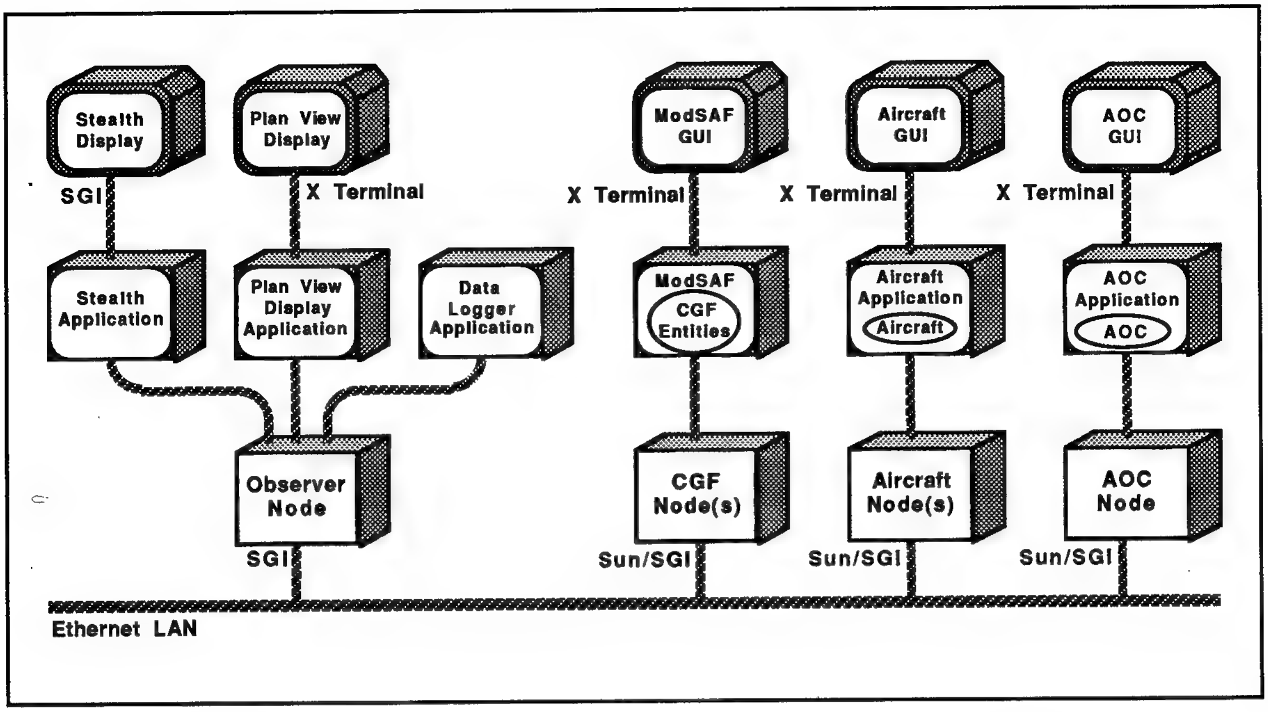

Figure 3-1. DIS for Tactical C³I System Configuration

In some cases, multiple DIS applications can be run on the same workstation or server, depending on the source of the applications (GOTS, COTS, or developmental) and how they interface with the network. Each copy of the CGF Node application, and the Observer Node Plan View Display application, which are both based on ModSAF, must be run on separate systems. Up to three separate applications which use VR-Link™ as the network interface, which include the Observer Node Stealth Vehicle Display and Data Logger applications, the Aircraft Node application, and the AOC Node application, can be run on an SGI workstation, using the VR-Link™ Packet Server, which interfaces to the network on behalf of the applications and allows them to both "share" the PDUs incoming from the network, and to each receive copies of the PDUs that the other applications using the Packet Server have transmitted. Because the version of the VR-Link™ Packet Server that was delivered with the demonstration software system is, like the Stealth Vehicle Display and Data Logger applications, specific to the SGI, only a single VR-Link-based application can be run on a Sun 4 workstation or server. VR-Link™-based applications and ModSAF-based applications cannot be run on the same workstation or server, as they each require exclusive control of the network interface.

Table 3-1. System Configuration

Node	Platform	Path	Executable
Observer Node - Plan View Display	SGI, Sun	DIS/ModSAF_1.4/common/src/ ModSAF	modsaf_sgi_5_2 modsaf_sun4
Observer Node - Stealth Vehicle	SGI	DIS/VR-Link_2.4.0/irix5/bin3	Stealth
Observer Node - Data Logger	SGI	DIS/VR-Link_2.4.0/irix5/bin3	xlogger
Computer Generated Forces (CGF) Node	SGI, Sun	DIS/ModSAF_1.4/common/src/ ModSAF	modsaf_sgi_5_2 modsaf_sun4
Aircraft Node	SGI, Sun	DIS/DIS_TC3I/bin/irix5 DIS/DIS_TC3I/bin/sun4	AIRCRAFT
Air Operations Center (AOC) Node	SGI, Sun	DIS/DIS_TC3I/bin/irix5 DIS/DIS_TC3I/bin/sun4	AOC

Table 3-1 summarizes the platforms on which each DIS application will run, the pathname of the directory where the application is located, and the name(s) of the executable file(s) for each application.

The graphical user interfaces of all of the DIS applications, except for the Stealth Vehicle Display, can be run remotely on any terminal, personal computer, or workstation that is capable of supporting an X Windowing System server. When the VR-Link Packet Server is used, this allows more applications to be run simultaneously than there are workstations or servers available. Each of the multiple applications running on top of the Packet Server can have its graphical user interface displayed on a different remote system. It should be noted however that X Window protocol traffic on the network will compete with DIS PDU traffic, possibly impacting performance.

3.2. OBSERVER NODE

The Observer Node consists of three independent DIS applications: the Plan View Display application, which is described in Section 3.2.1; the Stealth Vehicle Display application, which is described in Section 3.2.2; and the Data Logger application, which is described in Section 3.2.3.

3.2.1 PLAN VIEW DISPLAY

The Plan View Display application provides a two-dimensional map display showing the ground truth locations and movements of all simulated entities, as well as events such as weapons fire, detonations, and collisions.

The Plan View Display application is based on the Modular Semi-Automated Forces (ModSAF) system, which was developed by Loral Advanced Distributed Simulation under the Advanced Distributed Simulation Technology (ADST) contract for US Army STRICOM and ARPA. A copy of ModSAF can be run in a "SAFstation" mode, in which it does not simulate any entities, but simply processes the PDUs that it receives from the network and displays the locations and states of the reported entities. ModSAF is also the basis for the CGF Node, and so is discussed in more detail in Section 4.3.

3.2.2 STEALTH VEHICLE DISPLAY

The Stealth Vehicle Display application simulates an invisible observation vehicle which can be positioned anywhere within the synthetic environment (i.e., a flying carpet). This simulated stealth vehicle can be attached to any other entity, or group of entities, in a DIS exercise.

The Stealth Vehicle Display application is a commercial off-the-shelf (COTS) application developed by MaK Technologies. It is based on SGI's IRIS Performer 3D real-time display toolkit, and uses MaK's VR-Link™ toolkit to provide its DIS network interface. It runs only on SGI workstations.

The Stealth Vehicle Display application supports a number of different view modes, which determine how the viewpoint is controlled by user input, and/or by the positions and orientations of one or more simulated entities. The view modes supported by the Stealth Vehicle Display application are summarized as follows:

Figure 3-2. Stealth Vehicle Display

- **Absolute Mode** – This is the free-fly mode in which the Stealth Vehicle is not attached to any simulated entity. The arrow keys can be used to yaw and pitch the Stealth Vehicle, while the numeric keys can be used to move the Stealth vehicle forward/backward, left/right, and up/down.
- **Track Mode** – The Stealth Vehicle is not attached to any entity, and may be moved as in Absolute Mode, but the orientation of the Stealth Vehicle is constrained to automatically keep the tracked (secondary) entity in the center of the field of view.
- **Tether Mode** – The Stealth Vehicle is attached to a specific (primary) entity, but orientation can be manually controlled using the keyboard.

- **Tether-Track Mode** – The Stealth Vehicle is tethered to a specific (primary) entity, while its orientation is constrained to automatically keep the tracked (secondary) entity in the center of the field of view.
- **Compass Mode** – The Stealth Vehicle is attached to a specific (primary) entity, while its orientation is constrained to automatically keep the attached entity in the center of the field of view. The manual controls can be used to move the Stealth Vehicle around the attached entity in spherical coordinates.
- **Mimic Mode** – The Stealth Vehicle's position and orientation is attached to a specific (primary) entity. Manual controls move the eyepoint in the attached entity's body coordinate frame, giving the effect of being in the attached entity's cockpit.
- **Turret Mode** – Turret mode is similar to mimic mode, but instead of being attached to the body of the entity, the Stealth Vehicle is attached to the first articulated part of the entity. This is mainly useful for attaching to the turret of a tank.
- **Mimic-Track Mode** – The Stealth Vehicle's position is attached to a specific (primary) entity, with the orientation constrained to automatically keep the tracked (secondary) entity in the center of the field of view. The effect is that of being in the attached entity's cockpit, while tracking another entity.
- **Orbit Mode** – The Stealth Vehicle is attached to a specific (primary) entity, while its orientation is constrained to automatically keep the attached entity in the center of the field of view. The manual controls can be used to move the Stealth Vehicle around the attached entity in spherical coordinates.
- **Group Mode** – The Stealth Vehicle's position and orientation is automatically constrained so that all of the members of the specified group of entities are within the field of view.
- **Wide-Group Mode** – The Stealth Vehicle's position and orientation is automatically constrained so that all of the members of the specified group of entities are within the field of view, with the orientation chosen to maximize the apparent separation of the entities in the group.

3.2.3 DATA LOGGER

The Data Logger application records and plays back the network traffic of a DIS exercise in the form of a sequence of DIS Protocol Data Units (PDUs). In recording mode, the Data Logger application receives PDUs from the network and records them in a file. Once the network traffic of an exercise has been recorded, it can be played back. In playback mode, the Data Logger application reads recorded PDUs from a file and broadcasts them over the network, where they may be received by other DIS applications. The receiving applications cannot distinguish PDUs coming from the Data Logger application from the PDUs generated by other DIS applications.

The Data Logger application is a commercial off-the-shelf (COTS) application developed by MaK Technologies. It uses MaK's VR-Link™ toolkit to provide its DIS network interface, and has a Motif-based graphical user interface that resembles a VCR. It runs only on SGI workstations.

3.3. COMPUTER GENERATED FORCES (CGF) NODE

The Computer Generated Forces (CGF) Node application simulates a wide variety of friendly and enemy ground and air forces, as individual platforms and/or small units (platoons, companies, and flights). These simulated entities can be given tasks to perform, and will detect and respond to one another, and to entities simulated by other DIS applications, with a variety of movement, combat, and other types of behaviors.

The CGF Node application is a government off-the-shelf (GOTS) application which consists of the Modular Semi-Automated Forces (ModSAF) system. ModSAF was developed by Loral Advanced Distributed Simulation under the Advanced Distributed Simulation Technology (ADST) contract for US Army STRICOM and ARPA. ModSAF runs on SGI workstations under IRIX 5.2 or 5.3, or on Sun workstations under SunOS 4.1.3.

Multiple copies of ModSAF can be run simultaneously on different workstations as part of the same DIS exercise. The multiple copies communicate using a persistent object protocol, as well as through standard DIS PDUs, and will automatically balance the simulation load. The standard ModSAF configuration includes both the simulation application itself, and the Plan View Display which forms the user interface.

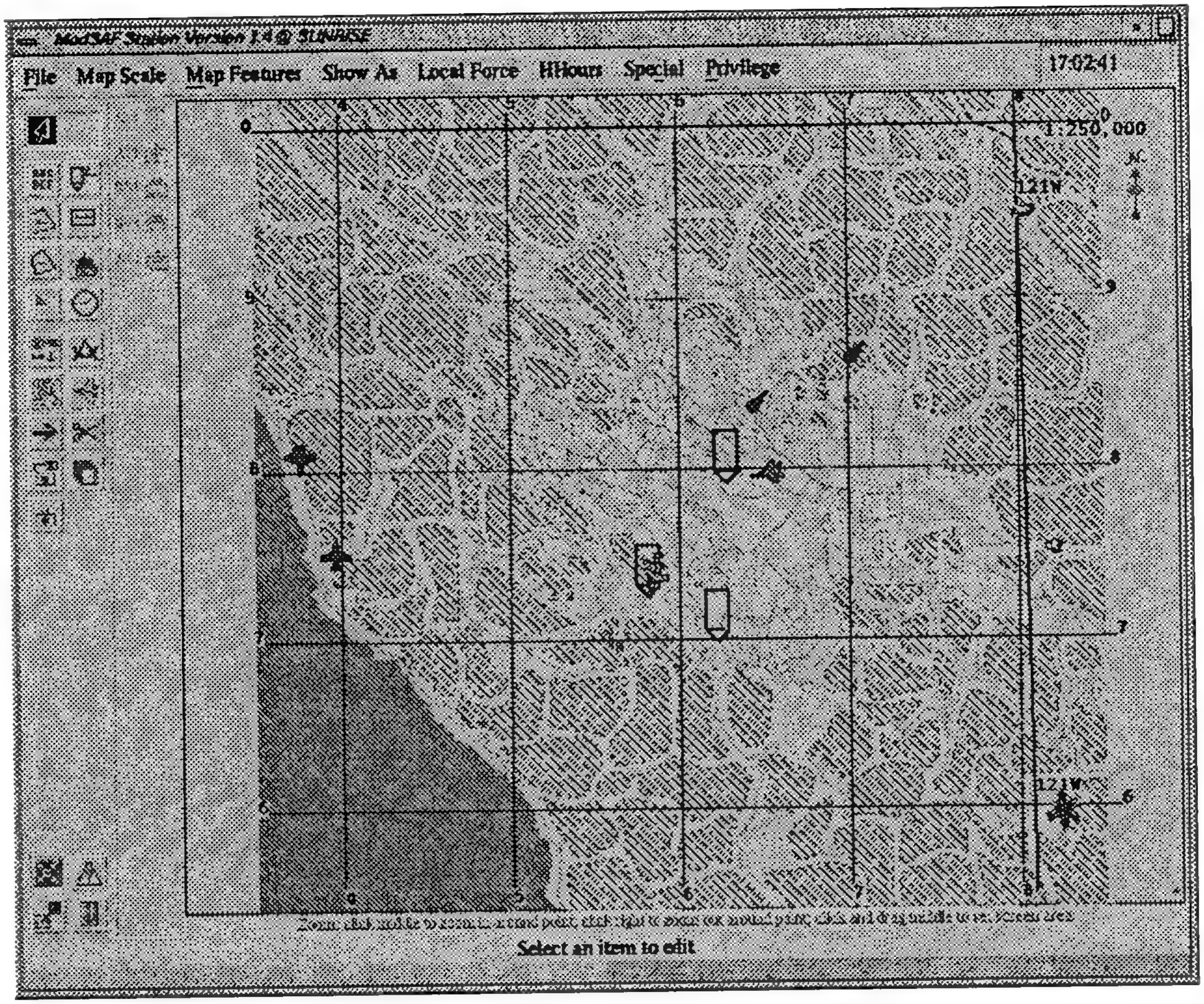

Figure 3-3. CGF/Plan View Display

These two components can also be run separately and independently. A "SAFsim" configuration includes only the simulation component, with no graphical user interface, and both generates PDUs for the entities that it is simulating, and processes received PDUs describing other entities in the exercise so that its entities can react to them. A "SAFstation" configuration provides a Plan View Display interface, as shown in Figure 3-3, and is the basis for the Observer Node Plan View Display application. It processes received PDUs and displays the locations and states of the entities and events that they describe, but does not generate PDUs.

The functionality provided by the CGF Node application includes:

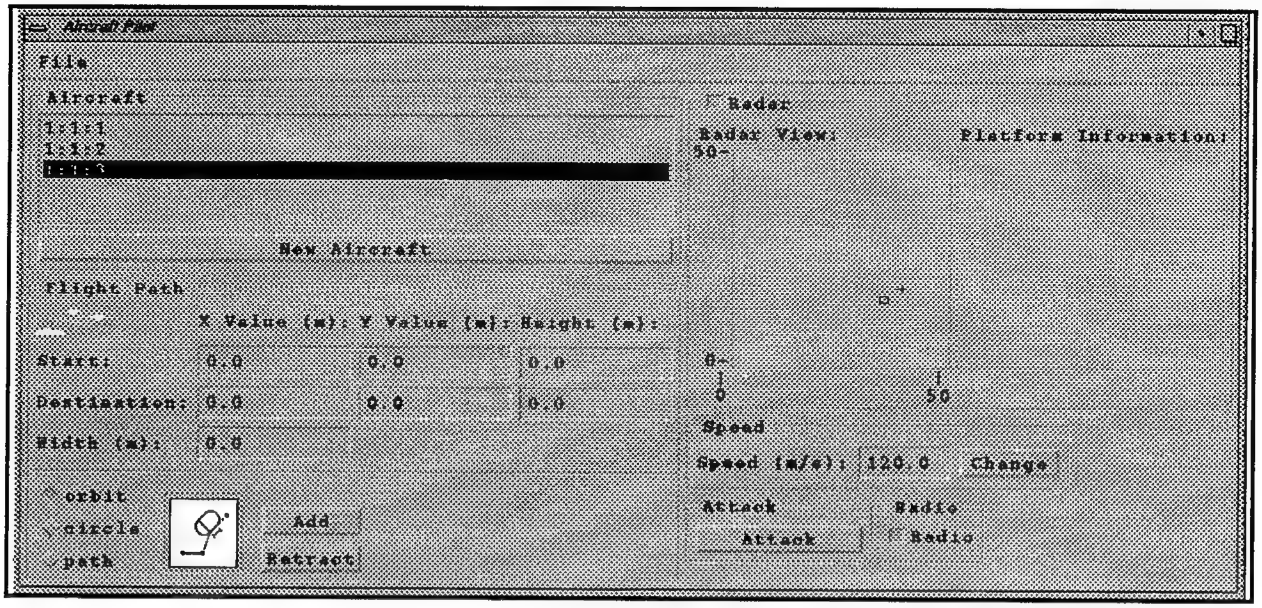

Figure 3-4. Aircraft Node Display

- the ability to manipulate the map display, including selecting various types of cartographic features and grid overlays to be included in the display, recentering the map display, changing its scale, and making various types of terrain measurements,
- the ability to annotate the map display with text messages and labels, point, line, and area graphics, and other special markers (e.g. minefield markers), organized into multiple overlays, and
- the ability to create, deploy, and task various types of military units consisting of ground vehicles (platoons and companies), helicopters and fixed wing aircraft (flights), dismounted infantry, and artillery, and to control their characteristics and rules of engagement.

3.4. AIRCRAFT NODE

The Aircraft Node is a DIS application which simulates one or more E-3 AWACS, E-8 JSTARS, F-15, and/or F-16 aircraft as DIS entities, including their associated radar and visual sensors, weapons, and communications capabilities. Each of the aircraft entities simulated by this application is capable of outputting and responding to DIS Entity State, Fire, Detonation, Collision, Transmitter, and Signal PDUs. Weapons flyout is simulated using temporary DIS entities representing fired missiles.

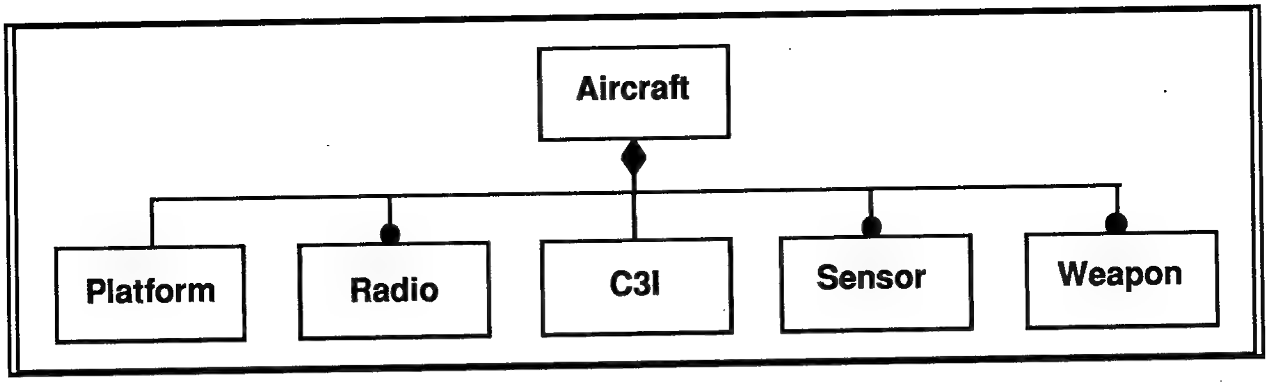

Figure 3-5. Aircraft Node Object Framework

As shown in Figure 3-4, the Aircraft Node application has a Motif-based graphical user interface, which displays the list of aircraft that the application is simulating, as well as the attributes of the currently selected aircraft and any targets which it has detected. The user interface allows the user to interactively control the simulated aircraft, including:

- creating a new aircraft of a specified type,
- specifying the destination of a selected aircraft, including options to orbit in a racetrack or circular pattern at a specified location,
- specifying the speed of a selected aircraft,
- turning the radio of a selected aircraft on or off,
- turning the radar of a selected aircraft on or off (aircraft visual sensors are always on), and for JSTARS, specifying the center of an area of interest,
- ordering an aircraft to attack a target at a specified location, which initiates an automated behavior sequence in which the aircraft flies to the specified location and circles it, repeatedly attacking any targets detected in that vicinity until they are destroyed, or until it has expended all of its munitions, and
- aborting an attack which is in progress.

All of the aircraft simulated by the Aircraft Node application report their detections of both ground and air targets, whether from their radar or visual sensors, to an Air Operations Center simulated by a copy of the AOC Node application. The simulated AOC can also transmit attack orders to the simulated strike aircraft.

The Aircraft Node application was developed by PGSC in C++, using algorithms, data structures, and other components from previous RL simulation efforts, repackaged within an object-oriented framework, as shown in Figure 3-5. Each aircraft object includes several component objects:

- a platform object, which defines the movement capabilities of the aircraft,
- zero or more radio objects, which provide the aircraft with communications,
- zero or more sensor objects, including radars and visual sensors, which provide the aircraft with detection capabilities,
- zero or more weapon objects, which provide the aircraft with attack capabilities,
- a C^3I object, representing the pilot/crew, which provides the automated attacking behavior of the aircraft, and which also serves as the interface through which the Aircraft Node application GUI controls the simulated aircraft.

The Aircraft Node application uses the VR-Link™ toolkit to provide its DIS network interface. The Aircraft Node application runs on SGI workstations under IRIX 5.2 or 5.3, or on Sun workstations under SunOS 4.1.3.

Multiple copies of the Aircraft Node application can be run simultaneously on the same workstation, or on different workstations, as part of the same DIS exercise. Running multiple copies of the Aircraft Node application on the same workstation requires the use of the VR-Link™ Packet Server. Typically, surveillance aircraft (E-3 AWACS and/or E-8 JSTARS) are simulated using one copy of the Aircraft Node application, while strike aircraft (F-15s and/or F-16s) are simulated using a separate copy. All of the aircraft simulated by a given copy of the Aircraft Node application report to a specific AOC, which is simulated by a specific copy of the AOC Node application.

3.5. AIR OPERATIONS CENTER (AOC) NODE

The AOC Node application is a DIS application which provides a very simple, abstract simulation of an Air Operations Center (AOC), including its own intelligence sources and communications. The AOC Node application is capable of outputting and responding to Transmitter and Signal PDUs, which represent the communications between the AOC and the friendly aircraft which report to it. To support this communication, the AOC is given a DIS entity identifier. However, the simulated AOC does not actually exist as an entity within the simulated environment.

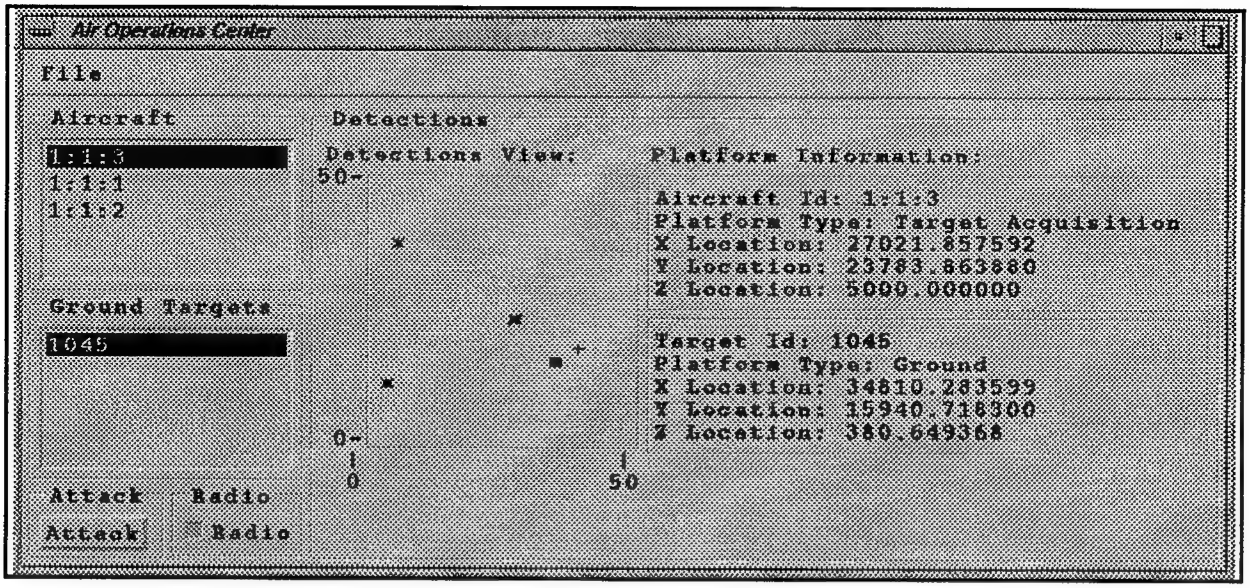

Figure 3-6. AOC Node Display

As shown in Figure 3-6, the AOC Node application has a Motif-based graphical user interface, which displays the list of aircraft that are currently reporting to the AOC, and a combined display of the aircraft and any targets reported by them, as well as a list of the ground targets reported to the AOC, the attributes of the currently selected aircraft, and the location of the currently selected ground target. The user interface allows the user to interactively perform the following operations:

- turning the AOC's radio communication on and off,
- ordering the currently selected aircraft to attack the currently selected ground target, which initiates an automated behavior sequence in which the aircraft flies to the specified location and circles it, repeatedly attacking any targets detected in that vicinity until they are destroyed, or until it has expended all of its munitions, and
- ordering the currently selected aircraft to abort an attack which is in progress.

The AOC Node application was developed by PGSC in C++, using algorithms, data structures, and other components from previous RL simulation efforts, repackaged within an object-oriented framework, as shown in Figure 3-7. Each aircraft object includes several component objects:

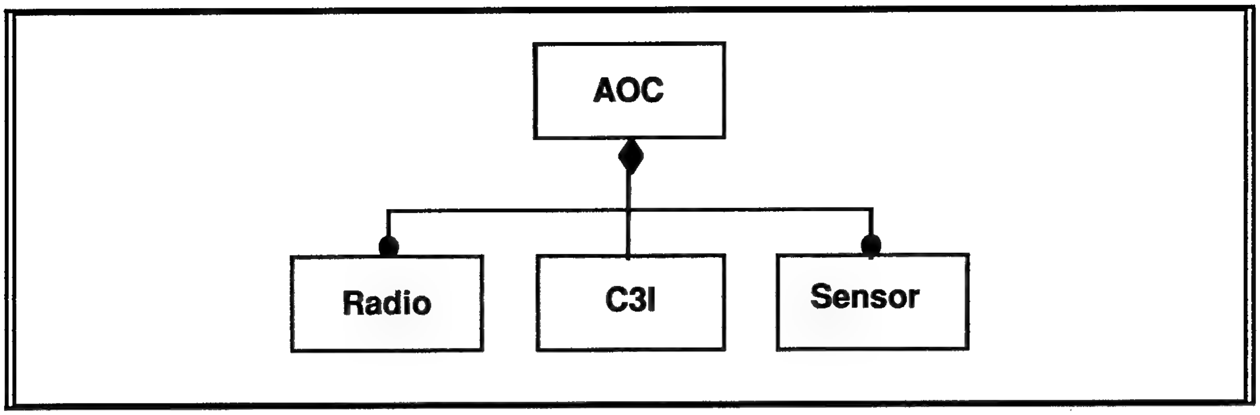

Figure 3-7. AOC Node Object Framework

- zero or more radio objects, which provide the AOC with communications capabilities,
- zero or more sensor objects, including radars and visual sensors, which provide the AOC with detection capabilities, and
- a C^3I object, representing the commander and staff of the AOC, which serves as the interface through which the AOC Node application GUI allows orders to be issued to the simulated aircraft controlled by the simulated AOC.

The AOC Node application uses the VR-Link™ toolkit to provide its DIS network interface. The AOC Node application runs on an SGI workstations under IRIX 5.2 or 5.3, or on a Sun workstation under SunOS 4.1.3.

Multiple copies of the AOC Node application can be run simultaneously on the same workstation, or on different workstations, as part of the same DIS exercise. Running multiple copies of the AOC Node application on the same workstation requires the use of the VR-Link™ Packet Server. Each copy of the AOC Node application receives target reports from one or more Aircraft Node applications, each simulating a set of one or more surveillance and/or strike aircraft.

3.6. DATABASES AND FILES

This section discusses the terrain databases and configuration files that are used by each of the DIS applications. Section 3.6.1 discusses terrain databases. Section 3.6.2 discusses the 3D models that support the Stealth Vehicle Display. Section 3.6.3 discusses the Aircraft and AOC Node configuration files.

3.6.1 TERRAIN DATABASES

The terrain databases delivered with the DIS for Tactical C^3I demonstration software system represent a 50km by 50km area of Fort Hunter-Liggett in California. Two different terrain databases for this area are used. The Stealth Vehicle Display application uses a terrain database in MultiGen® Flight format, which is optimized for visualization purposes. ModSAF, and the Aircraft Node and AOC Node applications, use a terrain database in the Compact Terrain Database (CTDB) format. Although these databases were created from the same source data, because they were created by different organizations using different processes they differ to some degree, which may cause anomalous behavior by both ground and air vehicles.

3.6.2 3D MODELS

The collection of 3D models delivered with the DIS for Tactical C^3I demonstration software system represents a variety of ground vehicles, aircraft, helicopters, and other objects. The Stealth Vehicle Display application uses these models, which are in MultiGen® Flight format, to support the visualization of the synthetic battlefield environment. The hierarchy of entity types defined by the enumerations associated with the DIS standard is mapped to a set of identifiers, and from these identifiers to specific model filenames, in the Stealth Vehicle Display application's configuration file. Note that models are not defined for all entity types, and in such cases, more generic models are used. For example, if a model is not available for a specific type of ground vehicle, the model associated with the generic ground vehicle identifier is used.

These 3D models were created by various organizations, and for a variety of different purposes. Most of them were obtained, directly or indirectly, from the Simulator Data Base Facility (SDBF) at Kirtland AFB. The models are not standardized with respect to origin location, orientation of local coordinate system, scale, color, or level of detail. Differences in origin location, orientation, and scale are compensated for by parameters in the Stealth Vehicle Display application's configuration files.

3.6.3 CONFIGURATION FILES

The Aircraft and AOC Node applications have been designed and implemented to be data driven to the greatest possible extent, in order to eliminate the need for recompilation and/or relinking each time the application's configuration is changed.

The Aircraft Node application configuration file contains the following information:

- the network UDP port number to be used for sending and receiving PDUs,
- the site, exercise, and application identifiers that are used to uniquely identify the Aircraft Node application in PDUs during DIS exercises,
- the time out interval, in seconds, to be used to time out remote entities for which Entity State PDUs are no longer being received,
- the maximum number of aircraft to be simulated by the Aircraft Node Application, used to allocate entity arrays,
- the reference latitude and longitude, in degrees, used to define the local Cartesian coordinate system,
- the mass of each aircraft type supported by the Aircraft Node application,
- information to be used in instantiating any new aircraft that are created interactively while the Aircraft Node application is running, including:
  - the default radio frequency and AOC site, application, and entity identifiers to be used by each aircraft type in reporting status and target detections,
  - the default radar parameters for each aircraft type, including antenna gain, bandwidth, power level, wavelength, minimum and maximum ranges, center azimuth and elevation (relative to the aircraft platform), and field of view width and height,
  - the default visual sensor parameters for each aircraft type, including minimum and maximum ranges, center azimuth and elevation (relative to the aircraft platform), and field of view width and height,
  - the default list of munitions carried by each fighter aircraft type, consisting of the number of munitions, followed by, for each munition, its DIS unique identifier, which includes domain, country, category, and subcategory fields, the speed of the munition, the fuse type of the munition, and the warhead type,
- the list of aircraft that should be instantiated when the Aircraft Node application is started, consisting of the number of aircraft, followed by, for each aircraft:

- the aircraft type code,
- the speed of the aircraft, in meters/second,
- the start and destination coordinates of the aircraft (in topographic coordinates – Northing, Easting, Down),
- the orbit length and width for the aircraft (aircraft are initially deployed orbiting in a racetrack pattern with the specified parameters),
- the height of the aircraft,
- the parameters of the aircraft's radar,
- the parameters of the aircraft's visual sensor,
- the list of munitions carried by the aircraft,
- the radio frequency and AOC site, application, and entity identifiers to be used by the aircraft type in reporting its status and any target detections,

The AOC Node application configuration file contains the following information:

- the network UDP port number to be used for sending and receiving PDUs,
- the site, exercise, and application identifiers that are used to uniquely identify the AOC Node application in PDUs during DIS exercises, and
- the radio frequency that is to be used for communication between the AOC and the aircraft that report to it, which is used in the Transmitter PDUs that represent this communication.

4. LIMITATIONS AND LESSONS LEARNED

This section identifies the most significant limitations of the DIS demonstration software system delivered under this contract, and also discusses the lessons learned during its assembly, development, and installation.

4.1 LIMITATIONS

The most significant limitations of the DIS demonstration software system delivered under this contract include:

- Only limited sets of aircraft, sensor, & weapon types are supported, and different sets of aircraft, sensor, and weapon types are supported by the Stealth Vehicle application, the CGF Node application, and the Aircraft Node Application.
- The number of entities that can be included in a DIS scenario is limited by the number of CPUs available, the power of each CPU, and network bandwidth.
- The performance of the Stealth Vehicle application display is limited by SGI Indy memory and graphics subsystem performance.
- The terrain database is limited to a 50km x 50km area at Ft. Hunter-Liggett, in California.

Each of these is discussed briefly below.

The DIS standards include the definitions of several collections of enumerated values which are used to identify simulated entities. These include country codes (which identify the country of origin of the design of a piece of equipment, rather than the country which owns a particular piece of equipment), domains (ground, air, space, etc.), and entity types (fighter) and subtypes (F-15, F-15E). The Stealth Vehicle application uses 3D polygonal models in MultiGen® Flight™ format to provide three-dimensional visualizations of entities in the synthetic battlespace. There currently is no standardized set of models for this purpose, although standards are being developed for several key aspects of 3D models, including the location of the origin of the model, the orientation of the model, the size of the model, how the model is placed on the terrain surface, and how attached and articulated parts are handled. These differences in 3D models are currently dealt with in the Stealth Vehicle application's configuration files. The models delivered under this effort were obtained from several

different sources, and vary considerably in quality. Some models were provided with the COTS VR-Link™ software, some were provided with the Fort Hunter-Liggett terrain database in Standard Simulator Data Base Interchange Format (SIF), and were converted to MultiGen® Flight™ format using GOTS software developed by the Institute for Simulation and Training (IST). They do not constitute a complete set of the entity types defined in the DIS standards. Similarly, ModSAF can be used to create a wide variety of entity types -- primarily ground vehicles, helicopters and aircraft -- but offers a limited set of entity types. This set can be expanded, but not easily. Finally, the Aircraft Node application is currently limited to surveillance aircraft (E-3 and E-8) and fighter aircraft (F-15 and F-16), but could be expanded to handle additional types relatively easily.

The number of entities that can be included in a DIS scenario with the demonstration system software is limited by the number of CPUs available, the power of each CPU, and the available local area network bandwidth. Each copy of ModSAF running on the network can generate up to 50-60 vehicles, organized into 10-12 platoons, depending on the power of the CPU on which it is running. Each copy of the Aircraft Node application can simulate up to several dozen surveillance and/or fighter aircraft, again depending on the power of the CPU on which it runs. In general, the DIS applications are limited more by the overall number of remote entities on the network, for which the application must process incoming Entity State PDUs. The bandwidth of an Ethernet-based LAN has been shown to be capable of supporting up to several hundred entities.

The display quality and performance of the Stealth Vehicle application is heavily dependent on the amount of memory available on the SGI workstation on which it runs, as well as on the capabilities of its graphics subsystem. The baseline SGI Indy installed in the ICARUS facility, with 32MB of memory and a baseline 8-bit XL graphics subsystem, will not in general be capable of updating the display in real time, even when texturing is turned off and there are only a small number of entities displayed on the terrain. This is because the system has insufficient memory to read in the entire terrain database. Colors will also be somewhat limited by the 8-bit graphics subsystem. The SGI Indy at PGSC's New Hartford facility, which has 96MB of memory and a 24-bit XZ graphics subsystem, has somewhat better performance, and is generally capable of updating the display in real time as long as texturing is turned off.

Support of real time texture requires a relatively high-end SGI workstation, such as an Onyx, or one of the new Indigo Impact workstations.

The DIS demonstration software system was delivered with only a single terrain databases, which represents a 50km by 50km area at Fort Hunter-Liggett on the coast of California. This is the database used to support the DIS interoperability demonstrations at the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) for the past several years. There are several reasons why only this one database could be delivered, including:

- Different terrain database formats are used by the Stealth Vehicle application and by ModSAF. The Stealth Vehicle application requires a terrain database in MultiGen® Flight™ format. ModSAF requires a terrain database to be in its own Compact Terrain Database (CTDB) format. For compatibility with ModSAF, the Aircraft Node application also uses the CTDB database format. While several different databases were available in each of these formats, the Fort Hunter-Liggett database was the only terrain database readily available in both formats.
- In addition to MultiGen® Flight™ and CTDB, DIS terrain databases are normally distributed in either Standard Simulator Database Interchange Format (SIF), or in Loral's S1000 format, which is derived from the original SIMNET terrain database format. However, software to convert terrain data between these formats and the target formats required by the Stealth Vehicle application and ModSAF was not available to PGSC.
- In general, terrain databases and database conversion software are not yet readily available throughout the DIS community. Several of the formats are vendor-controlled, and can be accessed only using API-based tools which are available only from that vendor. Standards for DIS terrain databases and associated software are being developed, but are still far from completion.

Although the terrain databases used by the Stealth Vehicle application and by ModSAF are derived from the same source data, the resulting databases were created using different processes, and therefore are not completely identical. For example, the elevation of the terrain surface at a given location may be different in the two databases. These differences can occasionally cause anomalous behavior,

particularly for low-flying aircraft that are trying to avoid the terrain. If the terrain used by the Stealth Vehicle application is higher than the terrain used by ModSAF and the Aircraft Node application, the aircraft may appear to fly through the terrain surface without colliding with it. If the reverse is true, the aircraft may crash while still appearing to be above the terrain surface. Ground vehicles may also appear to be either floating above the terrain, or partially embedded within the terrain surface.

4.2 LESSONS LEARNED

The lessons learned on this project can be summarized as follows:

- DIS is easy to do, especially when an off-the-shelf DIS networking package such as VR-Link™ is used, but is rapidly becoming more complicated.
- DIS is very demanding of hardware resources, including CPU power, graphics, memory, and disk space.
- Software development in a heterogeneous UNIX environment (SGI & Sun) using C++ is much more difficult than it should be.
- Using off-the-shelf software (COTS and/or GOTS) doesn't always make things easier, due to:
  - Frequent updates (VR-Link™, ModSAF), and
  - Poor documentation quality & support (CMTK, VR-Link™).
- Interfacing with developmental C^4I systems is very difficult (RAAP).

Building a DIS application, based on the current standards, is relatively simple. Currently, a basic DIS application needs to be able to respond appropriately to Entity State, Fire, Detonation, and Collision PDUs received over the network, and also needs to be able to output these PDUs when appropriate. Logistics PDUs are useful in some limited circumstances calling for refueling, rearming, or repair, but are not normally essential. Similarly, the ability to generate and respond to Transmitter and Signal PDUs is very useful, particularly in support of C^3I applications. The ability to support these PDUs will soon be required by any DIS application that simulates one or more entities that communicate with others. Support of the Simulation Management family of PDUs is not yet essential, but that may change very shortly. This will complicate DIS applications, as it will allow them to be managed by a remotely located Simulation

Manager. In general, as the number of different PDUs continues to grow, the complexity of a basic DIS application will continue to increase. Multicasting, which is expected to be added to the DIS standard in its next iteration, will significantly change the way in which applications interface with the DIS network. The new DoD High Level Architecture (HLA) initiative, which is intended to provide a framework for integrating simulation models at different levels of abstraction, will also complicate DIS applications, but will provide them with a more complete array of simulation support services over the network.

DIS applications are quite resource-intensive, in several different respects. No hardware was purchased under this contract. As a result, both in the developmental configuration at PGSC's New Hartford facility, and in the delivery configuration at Rome Laboratory's ICARUS facility, this project ran DIS applications on a variety of low-end Sun and SGI workstation configurations, with limited memory and disk space. For off-the-shelf components, such as ModSAF and the MaK Stealth Vehicle application, these configurations fell far short of the workstation configurations recommended by the developers. However, even these modest configurations were capable of running DIS applications with limited numbers of entities.

The development effort was complicated by the need to work within a heterogeneous UNIX development environment. Differences in the development tools available on different UNIX systems made the development process more difficult than it should have been, draining resources that should have been used to improve functionality. Due to the availability of better development tools, most of the development work on the Aircraft Node and AOC Node applications was performed on an SGI workstation. Once the applications had been tested in the SGI environment, they were then ported to the SunOS 4.1.3 environment. While most of the application components required only recompilation and relinking, differences in low-level system libraries required that several changes be made to the application source code. The heterogeneous environment also complicated the use of off-the-shelf software, as described below.

The use of off-the-shelf software components, both commercial (COTS) and Government (GOTS) under this effort, achieved mixed results. The use of MaK Technologies VR-Link™ DIS networking toolkit, Stealth Vehicle display, and Data Logger was very successful, although both the design of the toolkit and the quality of the documentation were disappointing. The use of ModSAF, which was developed by

Loral Advanced Distributed Simulation for STRICOM and ARPA under the Advanced Distributed Simulation Technology (ADST) program was very successful. However, the ModSAF software was found to be very complex and difficult to modify, and so was not used as the overall basis for the Aircraft Node or AOC Node applications, although the ModSAF library for accessing the terrain database (libctdb) was used in both of these applications. The attempt to use Rome Laboratory's Common Mapping Toolkit (CMTK) to provide background displays for the Aircraft Node and AOC Node applications was a failure, due to the size and complexity of the newer versions of the toolkit, which now has over 900 calls, and the poor quality of the documentation and support. Rather than saving time in the development effort, the attempted use of CMTK actually caused a significant number of labor hours to be wasted, as the integration of CMTK into the DIS applications could not be completed with the available resources.

Finally, interfacing a DIS network with an actual developmental C^4I system proved to be very difficult and ultimately infeasible within the scope determined by the available resources. It was originally intended to use the DIS network to drive the Rapid Application of Air Power (RAAP) system, but an investigation of the requirements of supporting such an interface determined that this would not be feasible. There were several reasons for this conclusion. First, RAAP is a theater-level system, and requires a theater-level scenario to drive its functionality properly. This could not be achieved with the hardware configurations available, which limited the number of entities that could be included in a scenario, or with the terrain databases available, which limited the physical extent of any scenario. Also, it would have been very difficult to provide RAAP with a complete and consistent set of data sources for the simulated scenarios, as RAAP relies on an intelligence database of fixed targets, as well as a map database of the corresponding area. Most importantly, however, it was discovered that RAAP current has no mechanism by which mobile, time critical targets could easily be reported to it, as it is designed to deal almost exclusively with various types of fixed targets, within the context of the current CTAPS system. Thus the decision was made to replace the originally planned RAAP interface with the AOC Node application, which simply accepts target reports and allows specific aircraft to be ordered to attack the detected targets. The AOC Node application remains a placeholder for the eventual integration of the DIS network with real C^3I systems.

5. CONCLUSIONS AND RECOMMENDATIONS

This section summarizes the potential benefits of the DIS technology demonstrated under this effort to Rome Laboratory and makes recommendations relative to how Rome Laboratory should continue to incorporate this technology into its C^3I system development efforts.

DIS technology has a number of potential benefits for Rome Laboratory, including:

- DIS technology can provide support for Rome Laboratory's contributions to the Time Critical Target problem, and specifically to the upgrading of the current CTAPS system to provide the ability to successfully prosecute Time Critical Targets.
- DIS technology can provide a common, distributed M&S infrastructure to support Rome Laboratory R&D programs in intelligence, surveillance, and C^3. This infrastructure, a local version of the "joint synthetic battlespace" from the Air Force's M&S vision, can be used to integrate many independent development efforts at Rome Laboratory, all of which use modeling and simulation technology to drive demonstration and testing of concepts, technologies, and systems. The interfacing of DIS-based simulation with actual C^4I systems is particularly important, allowing the value of improvements to existing systems, or the addition of new systems, to be demonstrated early in the development cycle.
- DIS technology provides opportunities for Rome Laboratory to increase its participation in the broader Air Force and DoD modeling and simulation communities, particularly in the integration of modeling and simulation with C^4I systems. As large-scale Air Force and Joint exercises become increasingly dependent on DIS technology, a DIS capability will become a requirement for participation.
- DIS technology can help to improve ties between Rome Laboratory and its customers, such as ESC, NAIC, & ARPA, through participation in distributed simulation exercises and experiments over the Defense Simulation Internet.

In order for Rome Laboratory to achieve the benefits identified above, the following actions are recommended:

- As the Air Force's laboratory for C^4I research and development, Rome Laboratory must get more involved in the use of advanced modeling and simulation technology, including:
  - Interfaces between simulations and actual C^4I systems, such as APS, RAAP, FLEX, etc.
  - High quality simulations of Air Force radars, sensors, intelligence systems, command and control systems, and communications systems, and
  - Environment databases, including terrain, atmosphere, and space.
- Rome Laboratory should establish an Advanced Distributed Simulation "Team", with representatives from the IR, C3, OC, and XP directorates. This team should develop contacts with Air Force and DoD modeling and simulation organizations including Headquarters US Air Force Modeling, Simulation, and Analysis Directorate (AF/XOM), the new Air Force Modeling and Simulation Organization (which is being created from AL/HLA-O), and the Defense Modeling and Simulation Office (DMSO). The members of this team should attend the DIS standards workshops and participate in all of the working groups and subgroups that are relevant to Rome Laboratory. In particular, Rome Laboratory personnel should participate in the DIS C^3I User Focus Group.
- Rome Laboratory should establish a "real" DIS facility, to serve as the hub of a local DIS network and as the gateway to the outside world through the Defense Simulation Internet. This facility should include a number of high-end graphics workstations and all of the hardware and software necessary to set up and execute relatively large-scale DIS exercises. The DIS demonstration software system delivered under this effort can serve as the starting point for such a facility. Rome Laboratory should pursue a full connection to the Defense Simulation Internet (DSI) through the Air Force's Advanced Connectivity Initiative, and should begin to acquire existing DIS-compatible models from other DoD organizations, and to make existing Rome Laboratory models and simulations DIS compatible.

☆U.S. GOVERNMENT PRINTING OFFICE: 1996-710-126-20138